Tìm Hiểu GIT

**1. Một số khái niệm.**

- Git là tên một hệ thống quản lý phiên bản phân tán.

- clone: mỗi 1 lần download dữ liệu về là 1 clone (một mã nguồn được nhân bản.)

- repository: kho chứa mã nguồn.

- commit: việc chỉnh sửa clone rồi cật nhật lại lên hệ thống.

**2. Cài đặt Git.**

- Link download: <https://git-scm.com/download/win>

- Lệnh tạo user name trên Git Bash: git config --global user.name “Loan Anh”

- Lệnh tạo user email: git config --global user.email “loananh30101995@gmail.com”

- Lệnh dùng để kiểm tra thông tin chứng thực: cat ~/ .gitconfig

**3. Tạo một repository.**

- Tạo local repository:

+ Cần truy cập vào thư mục của mã nguồn với lệnh: cd

+ Lệnh khởi tạo repository trong thư mục của mã nguồn, và đồng thời tạo ra một thư mục mới có thể chứa code sau này và khởi tạo repository cho nó : git init tên folder

+ Nếu kho chứa có sẵn mã nguồn thì phải đưa các tập tin về trạng thái Tracked bằng lệnh: git add tên\_file (có thể dùng \* để gom tất cả)

+ Lệnh để xem các tập tin đã được tracked: git status

+ Lệnh commit: git commit –m “Lời nhắn”

- Tạo repository trên Github:

+ Tạo New repository.

+ Clone kho chứa này về máy: git clone đường link

+ Truy cập vào thư mục vừa clone về: cd tên thư mục

+ Lệnh chỉnh sửa nội dung trong thư mục mình cần chỉnh sửa khi clone về: echo “# nội dung cần sửa” > tên thư mục cần sửa (có thể vô trực tiếp thư mục vừa clone về để chỉnh sửa)

+ Lệnh thêm thư mục cần đưa lên: git add tên thư mục

+ Thêm commit bằng lệnh commit

+ Lệnh push: git push origin master

\* Lưu ý: tất cả phải thực hiện ngay vị trí chứa thư mục đó.

**4. Tìm hiểu thêm về commit và staging area**

- Staging area là khu vực chuẩn bị cho quá trình commit. Muốn commit một tập tin nào thì tập tin đó phải nằm trong staging area.

- Để đưa một tập tin vào staging area dùng lệnh: git tênfile

- Muốn commit một tập tin, cần phải đưa tập tin đó vào trạng thái track bằng lệnh: git add tên file

\* Nhưng nếu tập tin đó đã track nhưng rơi vào trạng thái modified thì vẫn sẽ không thể commit được, mà phải đưa về staged bằng lệnh: git add

- Cũng có thể bỏ qua staging area để commit bằng lệnh: git commit –a –m “Lời nhắn”

- Tạo một tập tin (rỗng) mới trong thư mục làm việc bằng lệnh: touch tên file (tập tin vừa tạo sẽ ở trạng thái untracked)

- Lệnh xem trạng thái: git status

- Chuyển từ untracked về tracked bằng lệnh: git add tên file

- Chuyển từ tracked về untracked: rm tên file

- Bỏ tập tin ra khỏi staging area: git reset HEAD tên file

**5. Git log và undo commit**

**\*Git log**

- Để xem lịch sử các lần commit trước đó dùng lệnh: git log

- Để hiện thị chi tiết mỗi lần commit dùng lệnh: git log –m

\* Nếu log quá dài, dùng phím mũi tên lên xuống để đọc tiếp và ấn Ctrl + Z để thoát.

- Để xem 1 lần commit gần nhất thì thêm tham số -1 vào: git log -1

- Có thể sử dụng thêm một số tùy chọn xem log sau để tối ưu hơn quy trình đọc log:

--since, --after: Xem các lần commit kể từ ngày nhất định.

--until: Xem các lần commit trước từ ngày nhất định.

--author: Xem các lần commit của một người nào đó.(Vd: git log --author=user.name --pretty= “%s”)

--grep: Lọc các chuỗi trong log và in ra.

**-** Để lọc xem một đối tượng nào đó trong lịch sử commit dùng lệnh: git log –pretty=“%tag”

+ Danh sách các %tag:

* %H –  Commit hash
* %h – Abbreviated commit hash
* %T – Tree hash
* %t – Abbreviated tree hash
* %P – Parent hashes
* %p – Abbreviated parent hashes
* %an – Author name
* %ae – Author e-mail
* %ad – Author date (format respects the –date=option)
* %ar – Author date, relative
* %cn – Committer name
* %ce – Committer email
* %cd – Committer date
* %cr – Committer date, relative
* %s – Subject

**\* Undo commit**

- Xóa bỏ commit trước và commit lại dùng lệnh: git commit --amend –m “Lời nhắn”